

# EFFICIENTLY USAGE OF NOC ROUTERS BY USING BUFFERS

C.Suganya<sup>1</sup>, A.Bashilabanu<sup>2</sup>

<sup>1</sup>AssistantProfessor, <sup>2</sup>PG Student, Department of ECE, Bharath Niketan Engineering College ,(India)

## ABSTRACT

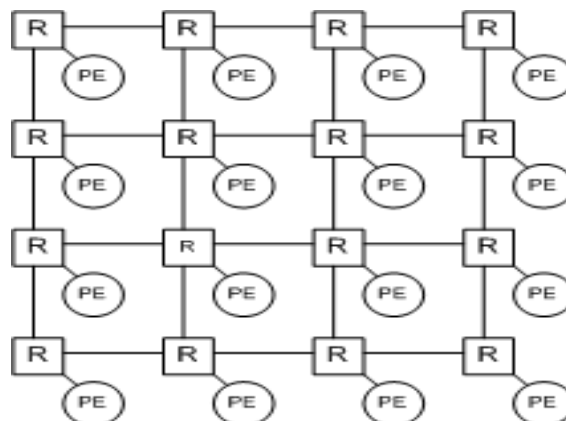
Router architecture plays a central role in the performance of a Network on Chip (NoC). Router consists of buffers which are dedicated to their input or output ports for temporarily storing packets during the times of congestion. Unfortunately, significant portion of the router area and power is consumed by the buffers alone. While running some tested traffic patterns, however, not all input ports of routers have incoming packets needed to be transferred simultaneously. Therefore, a large number of buffer queues in the network are empty whereas the other queues are mostly busy. This observation has led to the design of router architecture with multiple queue Virtual channel router which maximizes the buffer utilization by sharing the multiple buffer queues among input ports. Buffers become more efficient by sharing queues, hence the router is able to achieve higher throughput when the network load becomes heavy.

**Keywords:** IC, SoC, NoC, QoS, LRC, VC, WH.

## I. INTRODUCTION

### 1.1 Network Topology

On-chip networks share many concepts with an interconnection network for a traditional multiprocessor system. When we categorize networks, it is typically done by recognizing four key properties: topology, switching technique, routing protocol, and flow control mechanism. Mesh and torus network topologies are selected as the best choice in a NoC. These two network topologies have simplicity of 2-D square structure.



**Fig 1.1 Mesh Type Network Topology**

It is composed of a grid of horizontal and vertical lines with a router. This mesh topology is mostly used since delay among routers can be predicted in a high level. With an interconnection network for a traditional multiprocessor system. When we categorize networks, it is typically done by recognizing their key properties. A

router address is computed by the number of horizontal nodes and the number of vertical nodes. 2-D torus topology is a donut-shaped structure which is made by a 2-D mesh and connection of opposite sides.

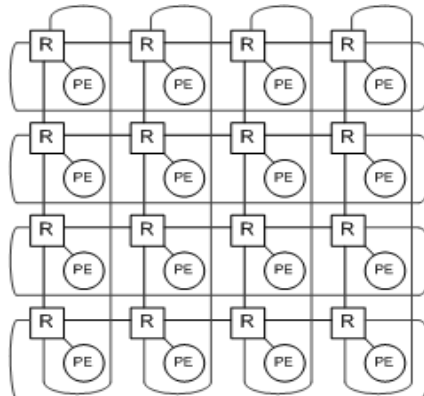


Fig. 1.2 Torus Network Topology

This topology has twice the bisection bandwidth of a mesh network at the cost of a doubled wire demand. But the nodes should be interleaved because all inter-node routers have the same length. In addition to the mesh and torus network topologies, a fat-tree structure is used.

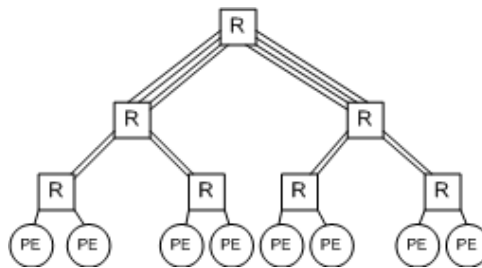


Fig. 1.3 Binary Fat Tree Network Topology

In M-ary fat-tree structure, the number of connections between nodes increases with a factor M towards the root of the tree. By wisely choosing the fatness of links, the network can be tailored to efficiently use any bandwidth. But the nodes should be interleaved because all inter-node routers have the same length. The number of connections between nodes increases with a factor M towards tree root.

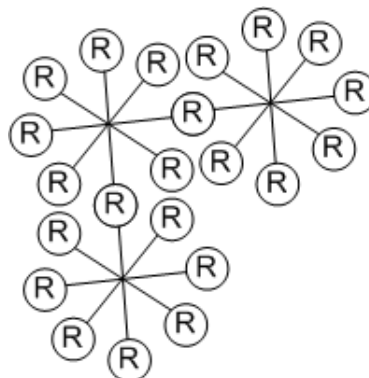


Fig. 1.4 Octagon Network Topology

Eight processors are linked by an octagonal ring. The delays between any two nodes are no more than two hops within the local ring. The advantage of an octagon network has scalability. For example, if a certain node can be operated as a bridge node, more Octagon network can be added using this bridge node. NoC topology and

characteristics are application-specific. NoC are mainly developed for consumer products so the constraints in area and cost are higher than in computer networks

## 1.2 Objective

The main objective of this project is to develop single queue based WH router and multiple queue based VC router. Also to design a chip level virtual 2x2 fashion router architecture and to compare the performance and arbitration time between both WH router based and VC based router. The main contributions of this paper are exploring and analyzing shared-queue router architectures that maximize buffer utilization for boosting network throughput and proposing a router architecture which allows input packet bypass shared queues for reducing zero-load and packet latency. Also evaluating and comparing the proposed router with WH routers in terms of latency and packet energy.

## II. EXISTING WORK

In the single router dead lock condition occurs, dedicated port may not be configurable for all the routers. In the wormhole switching method, the packets are split to Flow control digits (flits) which are snaked along the route in a pipeline fashion. Therefore, it does not need to have large buffers for the whole packets but has small buffers for a few flits. A header flit build the routing path to allow other data flits to traverse in the path. A disadvantage of wormhole switching is that the length of the path is proportional to the number of flits in the packet. In addition, the header flit is blocked by congestion, the whole chain of flits are stalled. It also blocked other flits. This is called deadlock where network is stalled because all buffers are full and circular dependency happens between nodes.

## III. PROBLEM STATEMENTS

Due to high latency the transmission is very slow .based on single queue the performance level is poor. There is a Dead lock problem which means if a packet at the head of a queue is blocked, all packets behind it stopped for transmission

## IV. PROPOSED WORK

To avoid the problems produced by the WH router .The VC router is implemented. The methods are given below,

### 4.1 Virtual Channel Router

This deadlock problem in WH router can be solved by a Virtual Channel router. The concept of virtual channels is introduced to present deadlock-free routing in wormhole switching networks. This method can split one physical channel into several virtual channels.

4.1.1.Virtual Channels

For real-time streaming data, circuit switching supports a reserved, point-to-point connection between a source node and a target node. Circuit switching has two phases: circuit establishment and message transmission. Before message transmission, a physical path from the source to the destination is reserved.

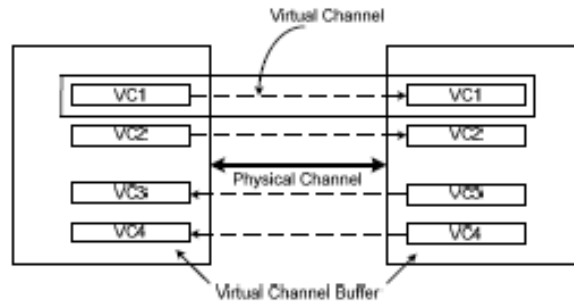


Fig.4.1.1 Concept of Virtual Channels

A header flit arrives at the destination node, and then an acknowledgement (ACK) flit is sent back to the source node. As soon as the source node receives the ACK signal, the source node transmits an entire message at the full bandwidth of the path. The circuit is released by the destination node or by a tail flit. Even though circuit switching has the overhead of circuit connection and release phase, if a data stream is very large to amortize the overhead, circuit switching will be used continuously. Since most Network-on-Chip systems need less buffering space and has a low latency requirement, the wormhole switching method with a virtual channel is the most suitable switching method.

4.1.2. VC Router Architecture

In this VC router design, an input buffer has multiple queues in parallel, each queue is called a VC, that allows packets from different queues to bypass each other to advance to the crossbar stage instead of being blocked by a packet at the head of the queue (however all queues at one input port can be still blocked if all of them do not win SA or if all corresponding output VC queues are full).

Because now an input port has multiple VC queues, each packet has to choose a VC of its next router's input port before arbitrating for output switch. Granting an output VC for a packet is given by a Virtual Channel Allocator (VCA); and this VC allocation is performed in parallel with the LRC; hence the router now has five stages as shown in below figure. Therefore, although a VC router achieves higher saturation throughput than a WH router while having the same number of buffer entries per input port, it also has higher zero-load latency due to deeper pipeline. Virtual channels allows adaptiveness in the network at the expense of buffer space and control logic so that the virtual channels can share the physical channels.

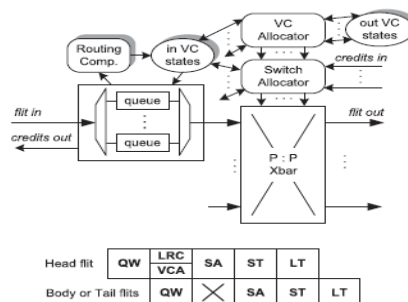


Fig.4.1.2 Five Stage Virtual Channel Router

Virtual channels share the same physical channel, but these virtual channels are logically separated with different input and output buffers.

## V. REALTIME APPLICATION

NoC are a key enabling technology for the provision of many additional services ranging from different Quality of Service (QoS) levels to fault-tolerance. Apart from global communications, the other major challenge facing designers now is high power dissipation. Power dissipation issues have grown to such importance that they now directly constrain attainable performance. Additionally, technology trends suggest that with further technology scaling communication power will demand an increasing proportion of the already limited system power budgets. For NoCs, it is now therefore important to understand any performance benefits they can deliver in the context of the power costs they demand.

- Networks enable the use of fault-tolerant wiring and protocols
- The network handles both pre-scheduled and dynamic traffic.
- Virtual channel routers are extensively used in Regulators (ViChaR – Virtual Channel Regulator) etc. A more real time example is shown below.

### 5.1. Example On-Chip Interconnection Network With Routers

To give a flavor for on-chip interconnection networks this section sketches the design of a simple network. Consider a 12mm x 12mm chip in 0.1mm CMOS technology with a 0.5mm minimum wire pitch. As shown in below Figure 4.1, we divide this chip into 16 3mm x 3mm tiles. A system is composed by placing client logic (e.g., processors, DSPs, peripheral controllers, memory sub systems, etc.) into the tiles. The client logic blocks communicate with one another only over the network. There are no top-level connections other than the network wires.

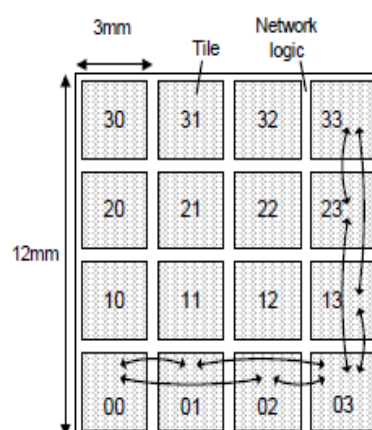


Fig.5.1 Partitioning the Die Into Network Logic

The network logic occupies a small amount of area between the tiles and consumes a portion of the top two metal layers for network interconnect. This baseline network uses a 2-dimensional folded torus topology with the nodes 0-3 in each row cyclically connected in the order 0,2,3,1. I/O pads may connect directly to adjacent tiles or may be addressed as special clients of the network

## VI. SOFTWARE DESCRIPTION

### 6.1 Software Used

The router logics that are discussed in chapter 3 are developed in Verilog language using portable GVIM portable editor and the simulation is carried out using Icarus Verilog compiler tool and waveform is analysed using gtkwave which comes as an in-built tool along with Icarus.

### 6.2 Information about Software

Basic information of the software used are given below

#### 6.2.1GVIM portable editor

Portable GVIM Easy version 7.4 editor is used for the module development. GVim Portable is a Microsoft Windows Application, therefore one need to have this operating system. VIM stands for ViIM proved .Most of Vim was written by Bram Moolenaar, et.al.

#### 6.2.2Icarus Verilog

Icarus Verilog version 9.7 compiler is used for the simulation of the verilog module developed using GVIM editor.Icarus Verilog can be simply termed as iverilog. iverilogis a compiler that translates Verilog source code into executable programs for simulation, or other net list formats for further processing. The currently supported targets are vvp for simulation, and FPGA for synthesis. Other target types are added as code generators are implemented.

### 6.3 Basic Encodings of Software

Few must to know encodings are listed below

#### 6.3.1GVIM portable editor

To start Vim, enter this command: >

```
gvim file.txt
```

In UNIX type this at any command prompt.

In Windows, open an MS-DOS prompt window and enter the command.

#### Inserting text

The Vim editor is a modal editor. The two basic modes are called Normal mode and Insert mode. In Normal mode the characters you type are commands.In Insert mode the characters are inserted as text. To start Insert mode type the "i" command (i for Insert).

#### Verilog Syntax selection

Select show file types in menu from syntax bar and select v>verilog HDL. This will take Verilog syntax format

#### Saving File

Once the modules are written and developed save the file with “.v” extension. Example:

```
File>Save as > *path*\file.v
```

#### 6.3.2 Icarus Verilog

To start compilation, change to the pathof the Verilog module file. Then enter this command: >

```
iverilogfile.v -o a.out
```

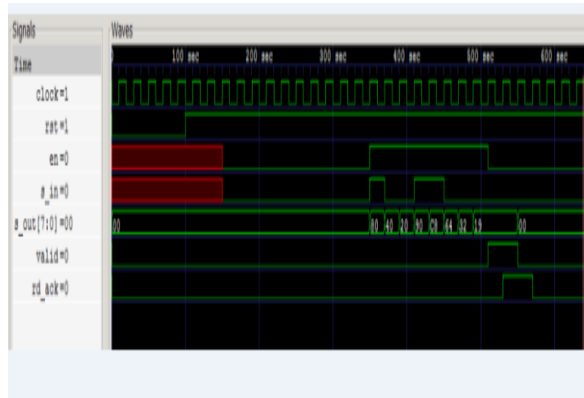
```
vvp.a.out
```

**gtkwavefile.vcd&**

Iverilog command will compile for errors and vvp will simulate and generate \*.vcd file which will be referenced for waveform opening in gtkwave waveform viewer.

**VII. SIMULATION RESULTS**

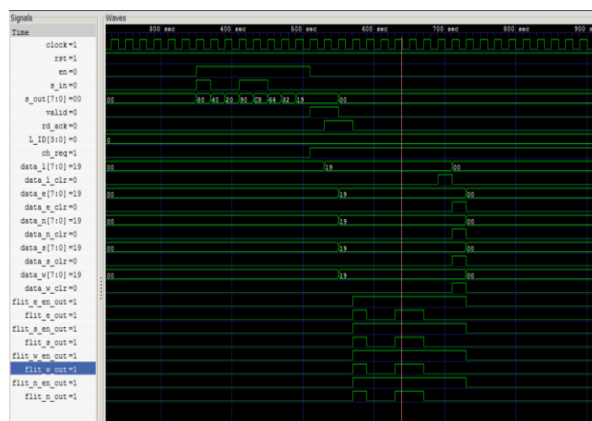
**7.1 Core 0 Buffer Write**



**Fig 7.1 Buffer Write Operation At Core 0**

Buffer block is designed and developed considering clock, rd\_ack, rst and enable signals as control signals, data input signals s\_in and output control signals as valid signals and data output signals as s\_out which is an 8 bit signal. Write operation will be performed, when en signal goes high. When en signal is high, the s\_in signal transfers the data into s\_out output signal. Once all the signal gets transferred, valid signal goes high. Also once the rd\_ack signal input has been received high, the valid goes low and all the data has been reset to initial values.

**7.2 Core 0 Local to All Data Transmission**



**Fig 7.2 Local To All Directions Data Transmission At Core 0**

Routers are designed in such a way that data transmission will be taken through all possible directions North, South, East and West. Apart from that there is local direction which stores the incoming data and transmits in all other directions. The data transmission happen when the enable signal is high. Once en signal is high the

incoming data gets transmitted to local direction then that will be transmitted towards data output in all the four directions.

### 7.3 Router Cross Bar Data Transmission

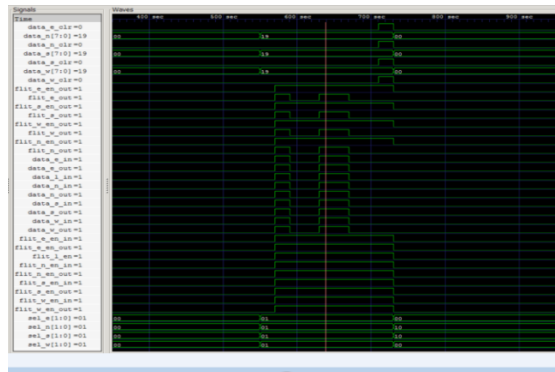


Fig 7.3 Router Cross Bar Signal Data Transmission

Router Cross bar section is developed in such a way that the packets that are received from north direction will be transmitted to south direction and vice-versa and information that are received from the east direction will be transmitted towards west direction and vice-versa. Apart from that this cross bar also transmits the information in local towards all four directions.

### 7.4 Router Controller Data Transmission

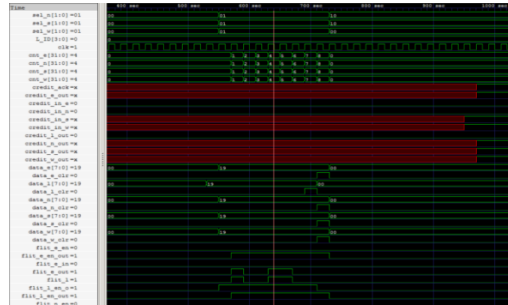


Fig.7.4 Controller Logic Signal Message Transmission

Router Controller section is developed in such a way that it majorly performs 3 operation. Where it responds to the incoming packets to the core and also transmits the packets that are received from north direction will be transmitted to south direction and vice-versa and information that are received from the east direction will be transmitted towards west direction and vice-versa. Controller also does the transmission of information in local towards all four directions which is shown in above simulation result.



### 7.5 Chip Level Flittransmission WH Router

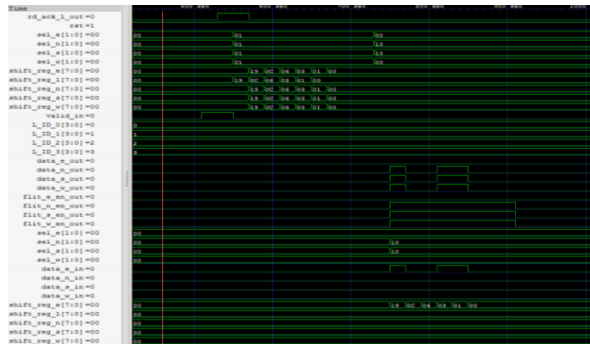


Fig 7.5 Core0 To Core1 Flit Transfer WH Router

Router design has been designed with 4 core blocks being core0, core1, core2 and core3. The information gets transmitted from core0 to core1, core2 and core3. The Flit transmission for core0 to core1 has been shown in the above simulated results.

### 7.6 Core0 to Core2 Flit Transmission WH Router

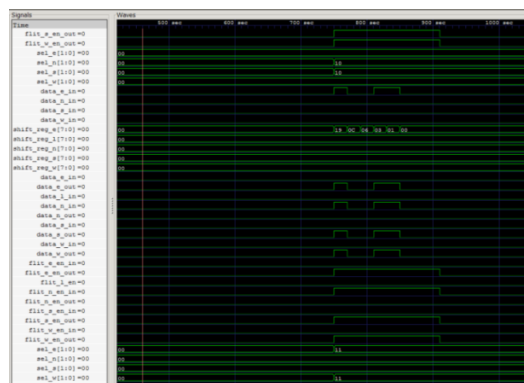


Fig 7.6 Core0 To Core2 Flit Transfer Wh Router

Router design has been designed with 4 core blocks being core0, core1, core2 and core3. The information gets transmitted from core0 to core1, core2 and core3. The Flit transmission for core0 to core2 has been shown in the above simulated results

### 7.7 Core0 to Core3 Credit Transmission Wh Router

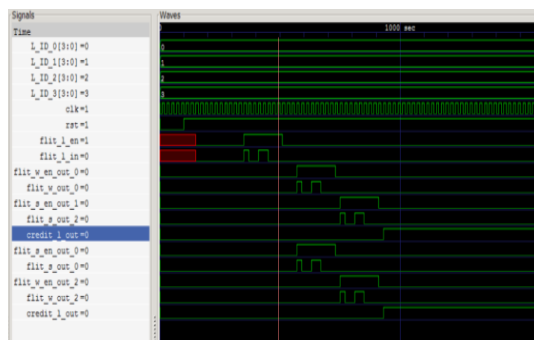


Fig 7.7 Core0 To Core3 Signal Gen Transmission Wh Router

Worm Hole Router design has been designed with 4 core blocks being core0, core1, core2 and core3. The information gets transmitted from core0 to core1, core2 and core3. The Flit transmission for core0 to core3 has

been shown generated. Once the enable signal is initiated, the input data will be transmitted which in turn triggers the credit signal generation. The credit signal generated results for core0 to core3 for wormhole router design has been shown in the above simulated results.

### 7.8 Virtual Channel Parallel Queue

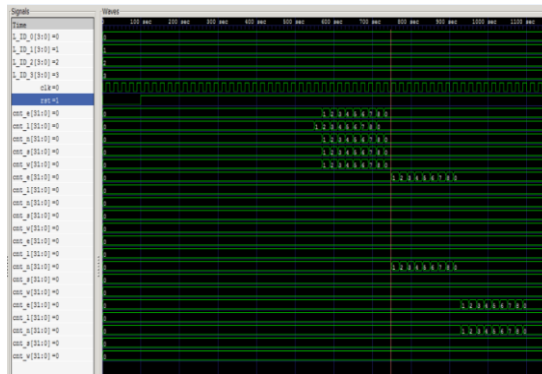


Fig 7.8 Vc Router Parallel Queue Data Transmission

Virtual Channel Router Architecture has been designed in such a way that the queue structure is designed parallel. Once the enable signal is initiated when one buffer is busy with holding data, the parallel buffer which will be free will take the data and once the cross bar is free enough to initiate the transmission the router initiates the data transmission .

### 7.9 Core 0 Flit Signal Transmission Vc Router



Fig.7.9 Core0 Flit Signal Transmission For A Vc Router

In a Virtual channel router, Buffer block is designed and developed considering clock, rd\_ack, rst and enable signals as control signals, data input signals s\_in and output control signals as valid signals and data output signals as s\_out which is an 8 bit signal. Write operation will be performed, when en signal goes high. When en signal is high, the s\_in signal transfers the data into s\_out output signal. Once all the signal gets transferred, valid signal goes high.

### 7.10 Core 0 Credit Signal Generation Vc Router

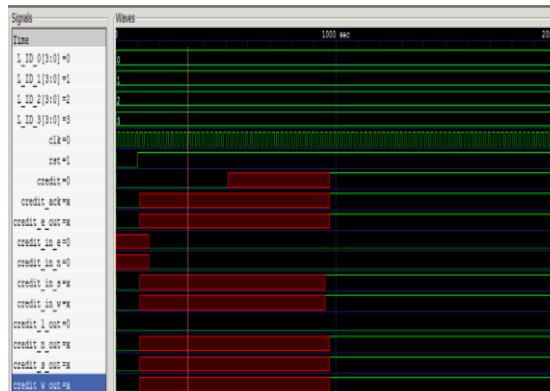


Fig.7.10 Core0 Credit Signal Transmission For Vc Router

Virtual Channel Router design has been designed with 4 core blocks being core0, core1, core2 and core3. The information gets transmitted from core0 to core1, core2 and core3. The Credit signal generation for VC router has been shown. Once the enable signal is initiated, the input data will be transmitted which in turn triggers the credit signal generation. The credit signal generated results for core0 credit signal generation for VC router for design has been shown in the above simulated results.

### 7.11 Core0 To Core3 Credit Transmission Vc Router

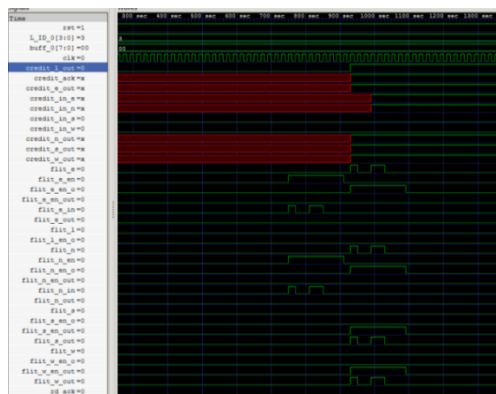


Fig 7.11 Core0 To Core3 Signal Gen Transmission Vc Router

Virtual Channel Router design has been designed with 4 core blocks being core0, core1, core2 and core3. The information gets transmitted from core0 to core1, core2 and core3. The Credit signal generation for VC router has been shown. Once the enable signal is initiated, the input data will be transmitted which in turn triggers the credit signal generation. The credit signal generated results for core0 to core3 for virtual router design has been shown in the above simulated results.

**Table 7.1 Latency Results Comparison**

<b>TRAFFIC PATTERN</b>	<b>WH ROUTER Latency (Cycles)</b>	<b>VC ROUTER Latency(Cycles)</b>
Buffer Write	12	8
Core0 to Core1	28	22
Core0 local to router	15	12
Core0 to core3	38	32
Credit Path	45	34

From the above table 6.1, we can observe that latency performance is compared between Worm Hole router and VC router whereas VC router achieves very less latency when compared to WH router. The clock cycle computation for the data valid signal from the clock cycle where the enable signal gets triggered is considered as the latency cycles which controls the speed of response for the particular router. So we can conclude that virtual channel router is benefited compared to worm hole routing.

### **VIII. CONCLUSION**

Network on Chip technology has been established as the preferred way of realizing system-level interconnect for most (if not all) high-end SoC products for nomadic and multimedia applications fabricated with the 45nm node. These SoCs embody either a proprietary NoC or a NoC from a third-party IP provider. While looking at current and future designs of complex SoCs in 45nm technology and beyond, NoCs are essential components to achieve performance and design closure. At the same time, there is a growing interest of using NoCs in complex FPGA designs as well.

The implementation of buffer module to store data packets, a Virtual channel Allocator (VA) and a Switch Allocator (SA), a crossbar switch in a generic router has been carried out and in addition to this control logic has also been implemented for both single queue based logic WH router and multiple queue based router VC router. From the results the deadlock problem that arise in the WH router has been rectified in VC router which shows better performance when compared to WH router. Also both the WH and VC routers are connected in arrayed router circuit and the implementation is carried out in 2x2 fashion array structure and the performance is analyzed and results shows that multiple queue logic router takes the advantage over WF router. The routers are reexamined through implementation using a hardware description language called Verilog. A Design Compiler is used to synthesize RTL (Register Transfer Level) code

## IX. FUTURE WORK

The future work can be extended with implementation of a novel router architecture that allowed sharing multiple 16X16 buffer queues for improving network throughput. Network on Chip technology has been established as the preferred way of realizing system-level interconnect for most Input packets also can bypass the shared queues to achieve low latency in the case that the network load was low. Also to find the best solution for dynamic buffer architecture. This buffer structure should avoid HoL blocking problem, increase buffer utilization compared to previous design, decrease arbitration time. The main contributions future work aims at exploring and analyzing shared-queue router architectures that maximize buffer utilization for boosting network throughput and proposing a router architecture which allows input packets to bypass shared queues for reducing zero-load and packet latency.

## REFERENCES

- [1] A. Banerjee, P. T. Wolkotte, R. D. Mullins, S. W. Moore, and G. J. M. Smit, "An energy and performance exploration of network-on chip architectures," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 17, no. 3, pp. 319–329, Mar. 2009.
- [2] A. Kumar, L.-S. Peh, P. Kundu, and N. K. Jha, "Towards ideal on-chip communication using express virtual channels," *IEEE Micro*, vol. 28, no. 1, pp. 80–90, Jan. 2008.
- [3] A. Prakash, "Randomized parallel schedulers for switch-memory-switch routers: Analysis and numerical studies," in *Proc. IEEE INFOCOM*, vol. 3. Mar. 2004, pp. 2026–2037.
- [4] C. H. V. Berkel, "Multi-core for mobile phones," in *Proc. DATE*, 2009, pp. 1260–1265.
- [5] D. Bertozzi, A. Jalabert, S. Murali, R. Tamhankar, S. Stergiou, L. Benini, and G. De Micheli, "NoC synthesis flow for customized domain specific multiprocessor systems-on-chip," *IEEE Trans. Parallel Distrib. Syst.*, vol. 16, no. 2, pp. 113–129, Feb. 2005.
- [6] G. De Micheli, C. Seiculescu, S. Murali, L. Benini, F. Angiolini, and A. Pullini, "Networks on chips: From research to products," in *Proc. 47th ACM/IEEE DAC*, Jun. 2010, pp. 300–305.
- [7] G. Michelogiannakis, D. Sanchez, W. J. Dally, and C. Kozyrakis, "Evaluating bufferless flow control for on-chip networks," in *Proc. 4<sup>th</sup> NOCS*, 2010, pp. 9–16.
- [8] J. Hu and R. Marculescu, "Energy-and performance-aware mapping for regular NoC architectures," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 24, no. 4, pp. 551–562, Apr. 2005.

## PROCEEDINGS PAPER:

1. Suganya.C,A.Bashila Banu "Deadlock Free Data Routing in 2x2 BufferSharing Routers",*Proc. International Conference current advancements in research and practises in emerging technology CARPET16.*